From: Greg Barnes
To: Microsoft ATR
Date: 1/22/02 12:05am
Subject: Microsoft Settlement

[Text body exceeds maximum size of message body (8192 bytes). It has been converted to attachment.]

January 21, 2002

To whom it may concern,

I am a software engineer in Seattle, Washington. I earned my Bachelor's Degree in Computer Science in 1987 from the University of California at Berkeley, and my Ph.D. in Computer Science and Engineering in 1992 from the University of Washington.

Upon reading the Proposed Final Judgment and the Competitive Impact Statement in the United States vs. Microsoft, it appears to me that the proposed final judgment is too lenient, too lax, and too full of loopholes.

The proposed final judgment is too lenient for Microsoft. They have been flaunting the law since they entered into a final judgment in 1995. The District Court found, and the Appeals Court agreed, that they have been illegally maintaining their monopoly for years, and in the process have essentially killed Netscape Corporation and threatened major computer companies such as Intel and Apple. It is widely believed that no venture capital money is available for a product that will or might compete with Microsoft, as Microsoft will crush any such competitor before it becomes to popular, whether by threats, or by folding software similar to the product into its operating system and distributing it 'free' to all who use their monopoly operating system.

Despite all this, the proposed final judgment does little to punish Microsoft for this conduct, or to insure that it cannot continue using the same tactics in the future. In particular, definition VI.U. says "The software code that comprises a Windows Operating System Product shall be determined by Microsoft in its sole discretion," thus making it impossible in the future to sustain a claim against Microsoft that it maintained its monopoly by illegally tying another product to its operating system.

I would like to see stricter conduct remedies, if not a structural remedy. I also share Ralph Nader and James Love's surprise (see http://www.cptech.org/at/ms/rnjl2kollarkotellynov501.html) that there is no monetary penalty attached, if only to prevent Microsoft from using its ill-gotten gains to buy or otherwise fund its way out of future competition.

The proposed final judgment is also much too lax. The judgment gives too much discretion to Microsoft, and its enforcement regime seems designed to allow Microsoft to delay its way out of any violations. The Technical Committee seems too powerless; it cannot impose any actual penalties, only report on violations, and the reports cannot even be made available to the public. Much of the apparent recent improvement in Microsoft's behavior appears to be due to publicity about its practices. It follows that the enforcement process should be as open as possible, not closed as the agreement stipulates.

Most disturbing, though, is the number of instances in the proposed final judgment where Microsoft is given the power to decide things on its own. The company has been found guilty of anti-trust violations, and has acted in bad faith towards the court both with regards to the earlier final judgment, and numerous times during the recent trial actions. Yet in many clauses, the judgment allows the company discretion to make exceptions. By past behavior, we can only expect the company to abuse this discretion, at the very least to delay any action it does not like, at worst to subvert the process.

For example, the 'notwithstanding' clauses after III.H.3 allow Microsoft discretion to void the previous section (and using the vague and undefined term 'reasonably prompt manner' to allow Microsoft to stall against any objections). Particularly troublesome is III.J.2, a clause which seems to be designed to allow Microsoft to ignore legitimate requests from

open source software developers (because they typically do not have a business behind them, and therefore no 'authentic' or 'viable' business). This is particularly troublesome because Microsoft has recently made a number of statements declaring open source software for example, 'a cancer', 'unamerican', and the number one threat to their business.

I would suggest that Nader and Love's suggestions about the term of the judgment and its enforcement mechanism be used. The current version merely seems to introduce a level of bureaucracy between Microsoft and actual enforcement power. In addition, instead of giving Microsoft latitude to make exceptions as in III.H.3 or III.J.2, Microsoft should be allowed to *propose* exceptions to the Technical Committee, which would make the final decision, with the stipulation that when in doubt, the committee should err on the side of openness to outsiders. Finally, open source software should be recognized as the public benefit that it is, and open source developers and projects be accorded the same privileges as standard businesses, if not more.

Finally, as an expert in the field, I would like to point out how riddled with loopholes the judgment is. The drafters seem myopically focused on the current characteristics of Middleware software and products, and unaware how simple it would be to circumvent the spirit of the judgment using its own text.

As a simple example, consider the various definitions of Middleware and Middleware Products. They all implicitly consider software that lies directly between the user and the operating system. It would be simple matter for Microsoft to break the agreement by devising another layer of middleware, between the current middleware and the operating system. For example, according to III.D., Microsoft must eventually release 'the APIs and related Documentation that are used by Microsoft Middleware to interoperate with a Windows Operating System Product'. With an extra layer added, currently existing middleware will no longer operate with the Operating System Product, and thus its APIs need not be released. I'm dubious the new layer, which does interoperate with the operating system, would be defined as middleware, either, as it would not seem to fit the judgment's definitions, which describe middleware in terms of current products.

Even if the enforcers of the judgment would not accept this transparent ploy, it seems likely to me that Microsoft could, as in the past, use their considerable political, legal and monetary power to delay a judgment against them. Given the short term of the judgment, this and similar ploys should be prohibited outright.

Let me give some more examples of how Microsoft could use the narrow language of the judgment to circumvent it. Section III.E. could be circumvented in a way similar to III.D, by imposing another layer between the server operating system and the Windows Operating System (say, another server operating system). The communications between the Windows Operating System and the first server operating system would have to be documented, but not the communications between the first server and the second. If this does not seem sufficiently devious, consider a protocol where the Windows Operating System sends 10 messages to (possibly different) servers, and the servers communicate between themselves to act on 1 of the 10 and ignore the other 9. Anyone who wishes to compete by supplying a substitute for the servers would be hard-pressed to figure out what was actually going on without information about the protocol between the servers, but this protocol need not be disclosed by Microsoft.

Note also that Microsoft's current direction (.NET) is toward Middleware accessing servers directly; it would seem by the judgment that Microsoft is free to abuse its monopoly as much as it wishes in this area, as the judgment does not place any conditions on middleware that does not talk to the desktop operating system.

Another avenue of circumvention is software such as Microsoft Office; clearly Office would not be defined as middleware or part of the operating system. However, there is nothing to stop Microsoft from grafting some communication code into future versions of Office to exploit these definitions. For example, suppose Microsoft includes code in its web browser that, on startup, communicates with a running Office program that it has started up, and the Office program in turn passes an 'OK to browse' message on to the Operating System. Suppose also that the Operating System code is rewritten so that, if a browser attempts to get a web page and the Operating System has not gotten such an 'OK', it terminates the browser program. None of the browser-Office or Office-operating system communication needs to be disclosed to outside parties, as Office is neither middleware nor part of the operating system. But the net effect would be that any non-Microsoft browser would be mysteriously broken.

Another problem arises with III.J.1.a, which allows Microsoft to refuse to disclose 'portions of APIs or Documentation or portions or layers of Communications Protocols the disclosure of which would compromise the security of a particular installation or group of installations of anti-piracy, anti-virus, software licensing, digital rights management, encryption or authentication systems.' This section can be abused to prevent disclosure of information about *any* protocol, by including in the protocol information which helps to disclose sensitive security information. Suppose, for example, that Microsoft devises two different versions of a Communications Protocol, and deems that it will use one version if the last bit of a site's secret cryptographic key is '0', and the other if the last digit is '1'. This means that if you know which protocol is being used, you know information that can help compromise this key (if you think one digit isn't enough information, suppose they have 1024 different versions, which will tell you 10 digits of the key, or 2^20 versions, which will tell you 20 digits, or whatever you like --- different versions could be as simple as including a different version number in the protocol's messages). The point is that, by III.J.1.a, Microsoft would be justified in refusing to document the Communication Protocol in question, as it would certainly compromise the security of the key used. And, again, this can be used to prevent disclosure of *any* protocol, whether related to security or not.

While these scenarios may seem far-fetched, we must consider past actions of the company. For example, when faced with an order to distribute a version of Windows '95 without Internet Explorer, Microsoft distributed a deliberately broken version. It is certainly not difficult to believe that the same company would resort to schemes such as these if it felt it would protect its illegally-maintained Windows monopoly.

I would urge the court to take action to prevent such loopholes. For example, it could hire a team of experts to look for any possible way the spirit of the agreement can be circumvented. The definitions, particularly those relating to middleware and operating system should be broadened. Finally, a list of underlying principles should be included with the judgment so that if any loopholes still remain, the Technical Committee and the Court will be able to disallow such evasions.

Sincerely,

Greg Barnes
gsbarnes@drizzle.com
4756 U Village Pl NE #239
Seattle, WA 98105